

Q1. Implicit Wave Propagation: Conjugate Gradients and Preconditioning.

A broken marble statue was reattached with a thick layer of soft epoxy filler, and conservators test the bond by sending a sound pulse through it. We model the path of the pulse as an elastic bar $x \in [0, 1]$ (marble for $x < \frac{1}{2}$, epoxy for $x > \frac{1}{2}$) whose displacement $u(x, t)$ satisfies the wave equation

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial x} \left(a(x) \frac{\partial u}{\partial x} \right), \quad a(x) = \begin{cases} 10, & x < \frac{1}{2} \text{ (stiff marble)}, \\ 1, & x > \frac{1}{2} \text{ (soft epoxy)}, \end{cases} \quad u(0, t) = u(1, t) = 0.$$

Introducing the velocity $v = \partial u / \partial t$ turns it into a first-order system.

Discretization.

- **Space:** split the bar into $N + 1$ equal pieces of length $h = 1/(N + 1)$, giving grid points $0 = x_0 < x_1 < \dots < x_N < x_{N+1} = 1$ with $x_i = ih$. The two end points are clamped ($u_0 = u_{N+1} = 0$), so only the N points inside the bar are unknown.
- **Time:** use time steps $t_n = n \Delta t$.
- **Vectors:** collect the displacements and velocities at the inside points into

$$\mathbf{u}^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_N^n \end{pmatrix} \approx \begin{pmatrix} u(x_1, t_n) \\ u(x_2, t_n) \\ \vdots \\ u(x_N, t_n) \end{pmatrix} \in \mathbb{R}^N, \quad \mathbf{v}^n = \begin{pmatrix} v_1^n \\ v_2^n \\ \vdots \\ v_N^n \end{pmatrix} \approx \begin{pmatrix} v(x_1, t_n) \\ v(x_2, t_n) \\ \vdots \\ v(x_N, t_n) \end{pmatrix} \in \mathbb{R}^N.$$

- **Stiffness:** let a_i be the value of $a(x)$ on the i -th piece $[x_{i-1}, x_i]$, for $i = 1, \dots, N + 1$ (so $a_i = 10$ in marble and $a_i = 1$ in epoxy). Then the term $\partial_x(a \partial_x u)$ is approximated inside at once by $-\frac{1}{h^2} \mathbf{K} \mathbf{u}$, where

$$\mathbf{K} = \begin{pmatrix} a_1 + a_2 & -a_2 & & & 0 \\ -a_2 & a_2 + a_3 & -a_3 & & \\ & \ddots & \ddots & \ddots & \\ & & -a_{N-1} & a_{N-1} + a_N & -a_N \\ 0 & & & -a_N & a_N + a_{N+1} \end{pmatrix} \in \mathbb{R}^{N \times N}$$

We step in time with

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \mathbf{v}^{n+1}, \quad \frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = -\frac{1}{h^2} \mathbf{K} \mathbf{u}^{n+1}.$$

This is called the **implicit Euler method**, which will be introduced later in the semester. Because the right-hand side is evaluated at the unknown level $n + 1$, every time step requires solving a linear system.

- (a) Eliminate $\mathbf{v}^{n+1}$ and show that one time step reads

$$\mathbf{A} \mathbf{u}^{n+1} = \mathbf{b}^n, \quad \mathbf{A} = \mathbf{I} + \frac{\Delta t^2}{h^2} \mathbf{K}, \quad \mathbf{b}^n = \mathbf{u}^n + \Delta t \mathbf{v}^n.$$

($\mathbf{A}$ is symmetric positive-definite, so CG applies.)

- (b) Take $N = 3$ ($h = \frac{1}{4}$, so $\mathbf{u} = (u_1, u_2, u_3)^\top$ lives at $x = \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$) and $\Delta t = \frac{1}{4}$. Write out $\mathbf{A}$.
- (c) (*Optional*) The bar is released from rest ($\mathbf{v}^0 = \mathbf{0}$) at the displaced state $\mathbf{u}^0 = (1, 1, 1)^\top$. Solve the first time step $\mathbf{A}\mathbf{u}^1 = \mathbf{b}^0$ with **conjugate gradients** starting from $\mathbf{x}_0 = \mathbf{0}$. Report the iterates $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$. (Hint: you may use CG code from class materials.)
- (d) (*Optional*) Compute the condition numbers $\kappa(\mathbf{A})$ and $\kappa(\mathbf{P}^{1/2}\mathbf{A}\mathbf{P}^{1/2})$ for the Jacobi preconditioner $\mathbf{P} = \text{diag}(1/A_{11}, 1/A_{22}, 1/A_{33})$ (you may use `np.linalg.eigvalsh`). What does this predict for preconditioned CG on a much finer grid?

Q2. Nonlinear Ultrasonic Response: Gauss-Newton and Levenberg-Marquardt.

Damaged or poorly bonded materials respond *nonlinearly* to strong ultrasonic excitation, and *nonlinear ultrasonic testing* uses this effect to detect weak bonds. We model it by adding a hardening restoring force $-\gamma u^3$ ($\gamma = 1600$) to the bar of **Q1**:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial x} \left(a(x) \frac{\partial u}{\partial x} \right) - \gamma u^3.$$

We use the same grid, vectors, and matrices as in **Q1(b)** ($N = 3$, $\Delta t = \frac{1}{4}$). With the cube taken entrywise, $\mathbf{u}^{\circ 3} \equiv (u_1^3, u_2^3, u_3^3)^\top$, the implicit Euler method reads

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \mathbf{v}^{n+1}, \quad \frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = -\frac{1}{h^2} \mathbf{K} \mathbf{u}^{n+1} - \gamma (\mathbf{u}^{n+1})^{\circ 3}.$$

Eliminating $\mathbf{v}^{n+1}$ as in **Q1(a)**, each time step now requires solving a *nonlinear* system:

$$\mathbf{G}(\mathbf{u}^{n+1}) = \mathbf{0}, \quad \mathbf{G}(\mathbf{u}) = \mathbf{A}\mathbf{u} + \mu \mathbf{u}^{\circ 3} - \mathbf{b}^n, \quad \mu = \gamma \Delta t^2 = 100,$$

with $\mathbf{A}$ and $\mathbf{b}^n = \mathbf{u}^n + \Delta t \mathbf{v}^n$ from **Q1**. We solve it by minimizing $E(\mathbf{u}) = \frac{1}{2} \|\mathbf{G}(\mathbf{u})\|_2^2$. Starting from a guess $\mathbf{u}^{(0)}$, each iteration updates $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \boldsymbol{\delta}$, where

$$\text{Gauss-Newton (GN): } \mathbf{J}^\top \mathbf{J} \boldsymbol{\delta} = -\mathbf{J}^\top \mathbf{G}, \quad \text{Levenberg-Marquardt (LM): } (\mathbf{J}^\top \mathbf{J} + \lambda \mathbf{I}) \boldsymbol{\delta} = -\mathbf{J}^\top \mathbf{G},$$

and $\mathbf{G}$ and the Jacobian $\mathbf{J}$ (the 3×3 matrix with entries $J_{ij} = \partial G_i / \partial u_j$) are evaluated at $\mathbf{u}^{(k)}$.

We take the first time step from rest, $\mathbf{u}^0 = (1, 1, 1)^\top$ and $\mathbf{v}^0 = \mathbf{0}$, with the equilibrium position $\mathbf{u}^{(0)} = \mathbf{0}$ as the initial guess.

(a) Write out $\mathbf{G}(\mathbf{u})$ and $\mathbf{J}(\mathbf{u})$ entry by entry.

(b) Compute one GN step, $\mathbf{u}_{\text{GN}}^{(1)}$. (Hint: at $\mathbf{u} = \mathbf{0}$, $\mathbf{J}$ is invertible, so the GN step reduces to $\mathbf{J}\boldsymbol{\delta} = -\mathbf{G}$. Which system from **Q1** is this?) Compare $E(\mathbf{u}^{(0)})$ and $E(\mathbf{u}_{\text{GN}}^{(1)})$. What went wrong?

(c) Compute one LM step with $\lambda = 10$, $\mathbf{u}_{\text{LM}}^{(1)}$ (you may solve the 3×3 system with your CG code from **Q1**), and compute $E(\mathbf{u}_{\text{LM}}^{(1)})$. Did the damping help?

(d) (*Optional*) Iterate both methods from $\mathbf{u}^{(0)} = \mathbf{0}$. For LM, halve λ after a step that lowers E ; otherwise reject the step and double λ . Plot $E(\mathbf{u}^{(k)})$ against k on a logarithmic scale, and compare the converged $\mathbf{u}^1$ with the linear result of **Q1(c)**.